From: Professor Zurab Bosikashvili

The Multi Agents Architecture Of The Reasoning System On Base Grid Computing

Tbilisi, Georgia, October 2014

AGENDA

- Overview
- Complex Problems
- Grid and Cloud Computing
- Multi Agents and Grid Computing
- Agent Model
- Reasoning Model
- Argumentation Reasoning Model

The Complex problems

- Big Multi-Dimensional Data processing
- Semantic Web
- Resources Distribution problems
- Reasoning Problem
- Graph Isomorphism Problem
- …

Have Polynomial or Exponential Complexity

Real Time Direct Solution

The Traditional Computing Architectures

Needs New Solutions

Grid and Cloud Computing

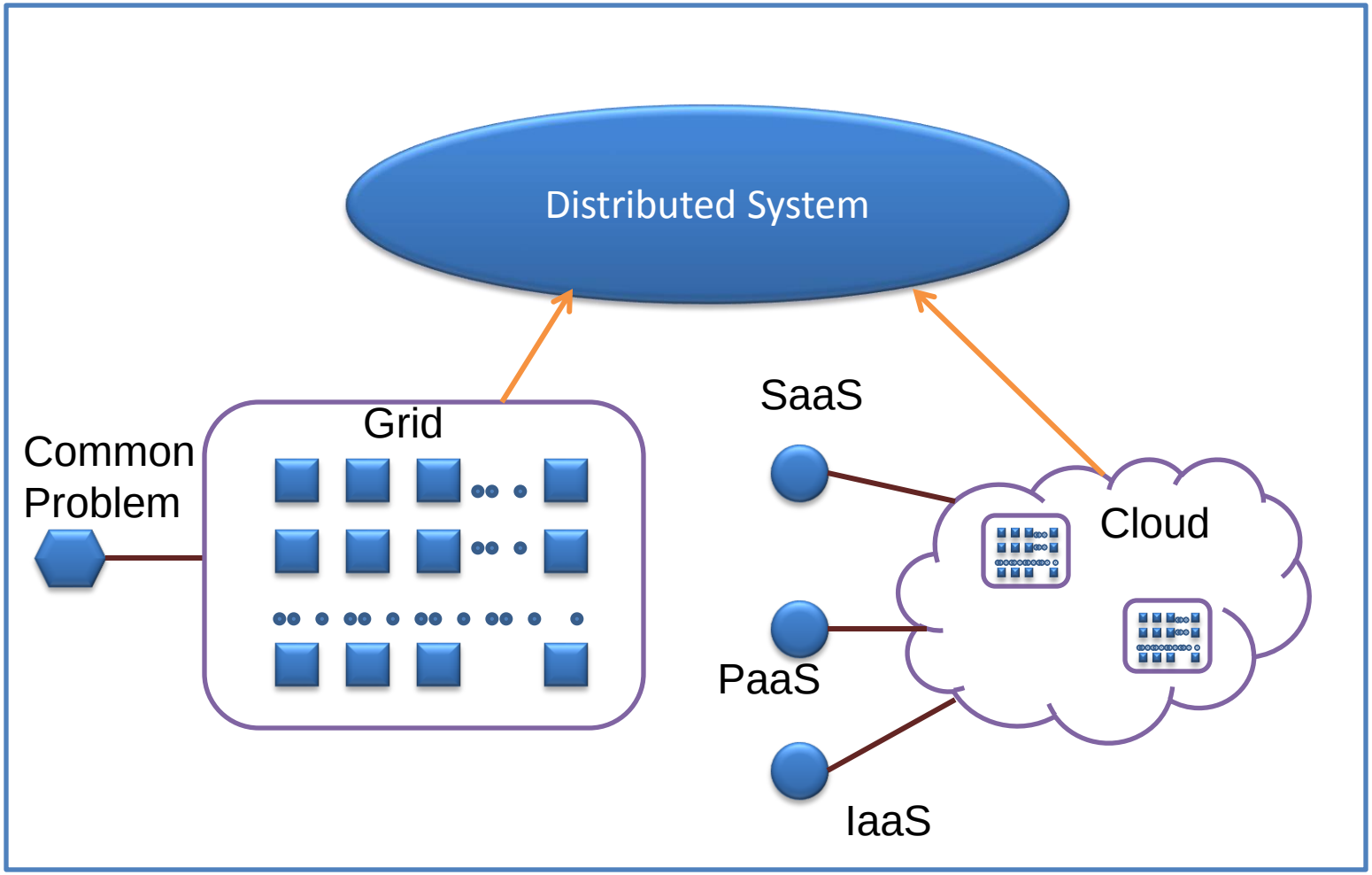
The grid computing is a form of data processing via a computer network. The main concept is that all computer's resources like processing power, memory and data storage are shared with the every computer in the system. The main goal is to force all computers to work for the benefit of one system. In the grid all connected computer's are seen as one powerful processing center.

The main issue that touch the grid computing is a coordination of shared resources, in means of direct access to the computer, software, data, and other resources to solve a problem (task). The problem rises when the system has to work in dynamic and heterogeneous environment.

For Cloud computing is a model of information processing based on services provided by (external or internal) provider. The resources should be provided as rapidly as is it possible, released without any unnecessary configuration and with minimal provider interaction.

With the subject of the cloud computing is related a lot of important issues. An organizations concerned about the future of this idea was gathered and discussed main problems.

Grid and Cloud Computing



The multi-agent approach

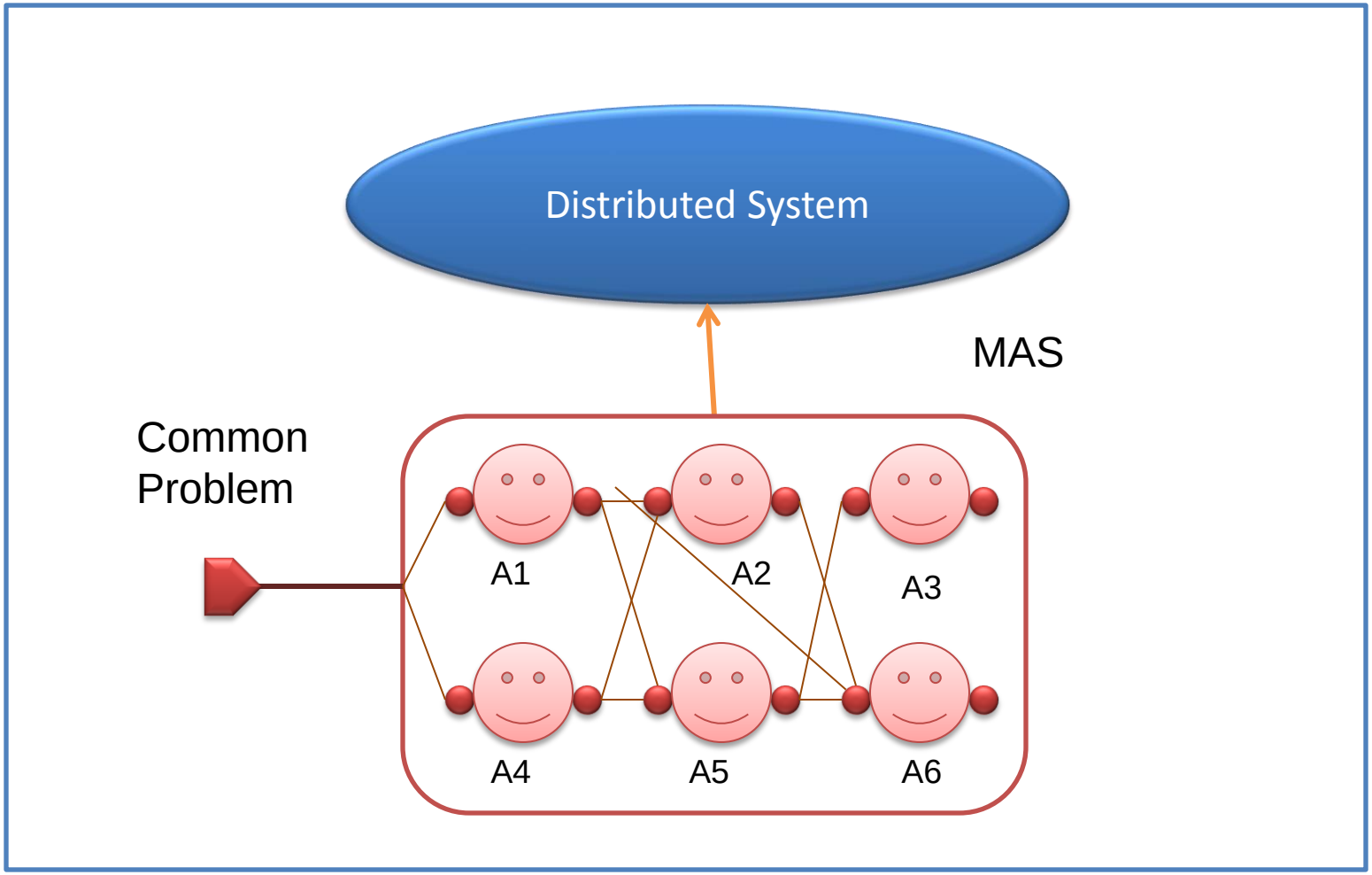
Using the multi-agent approach will have the following major advantages:

- It makes use of parallelism. For example, a goal can be sent to several agents so that they can try to solve it simultaneously. In the situation where the agent knowledge bases are big and disjunctive, this advantage is significant comparing to a single multithread agent with a single knowledge base.

- It makes hierarchical planning more efficient. For example, we can have a coordinate robot to find a top level plan comprises decomposable goals and then send the goals to different robots to solve.

In the cases where the knowledge base is not mergeable, the multi-agent approach is unavoidable and hence distributed reasoning is needed.

Multi Agents System (MAS)



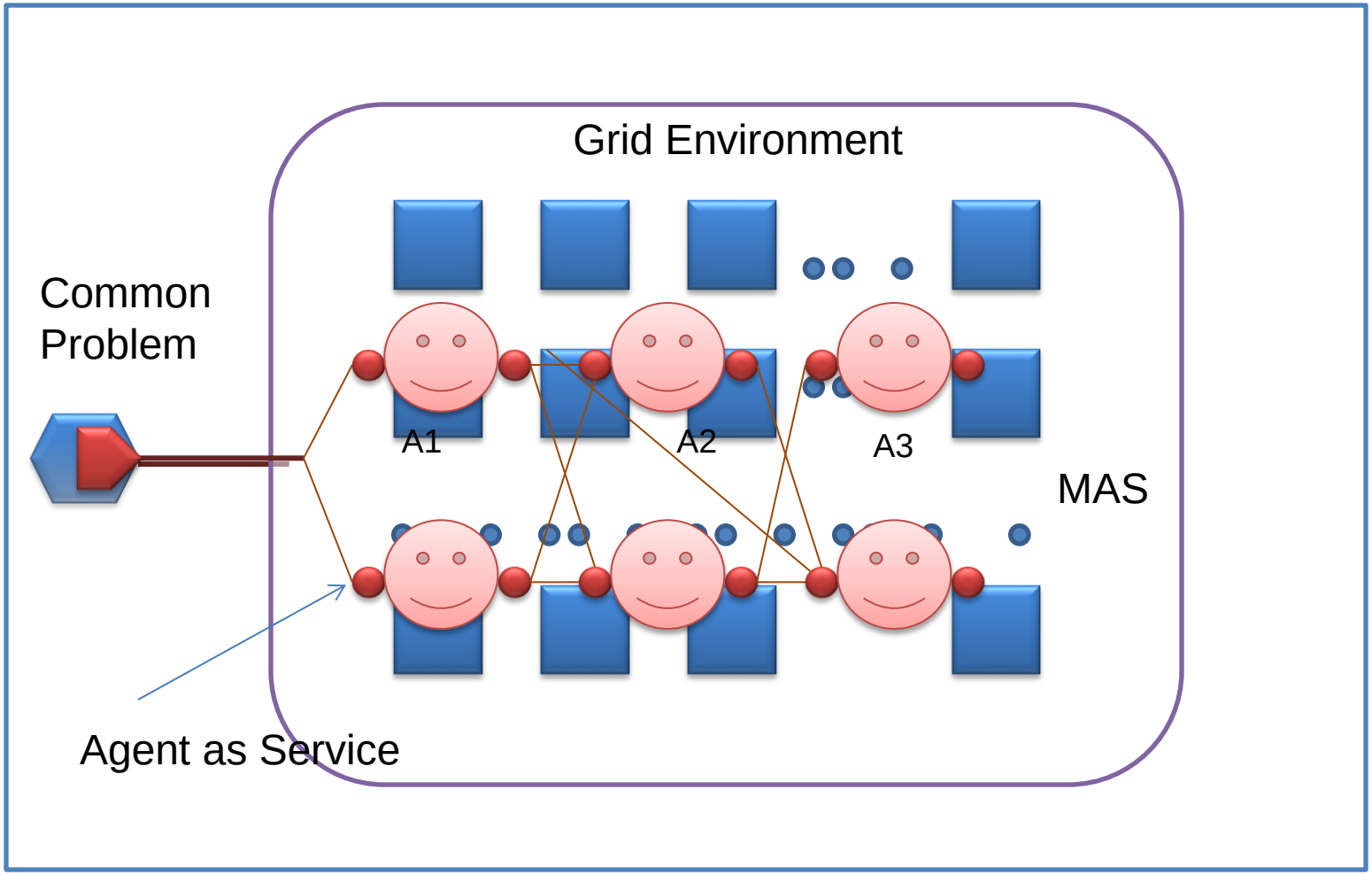
MAS + Grid + Distributed Reasoning

- The Grid systems used to be concerned as a systems infrastructures (tools and applications) for reliable and secure computer resource sharing in dynamic and distributed computer network in order to solving high demand computational problems

- In the same time MAS were explicit in the direction of solving problems that require autonomous and intelligent actions in flexible and uncertain environment

- Recently engineers began to notice the benefits of combination of those two systems.

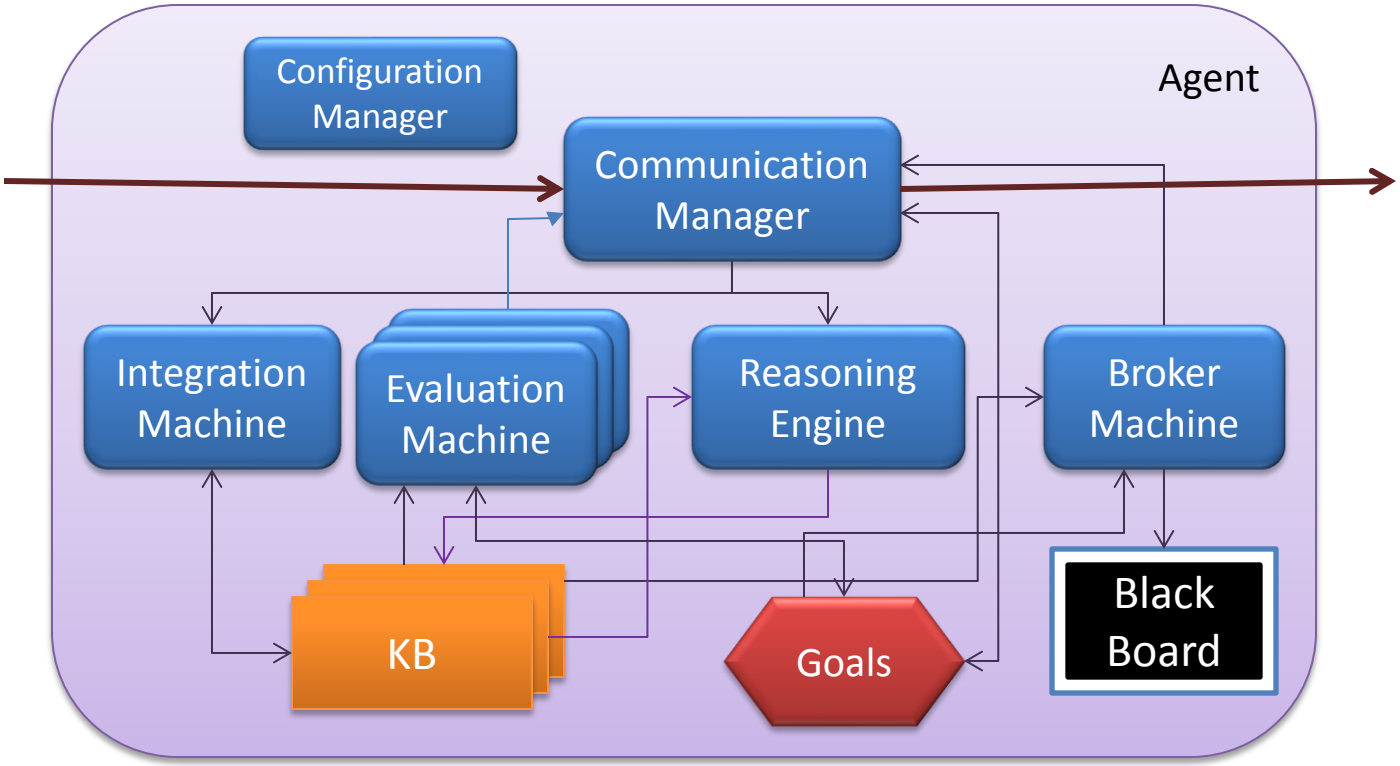
Integration MAS To Grid



The Issues of The Integration MAS Into Grid

- Improvement of agent architecture
- Consideration an agent and group of agents as a service
- New hierarchical classification schema of agents
- Transformation of agents interaction graph to the loosely coupled agents interaction graph
- Presentation MAS as a Stigmergic agent system (SAS)
- Distributed reasoning and Agent-based argumentation

The architecture of the Intellectual abductive agent



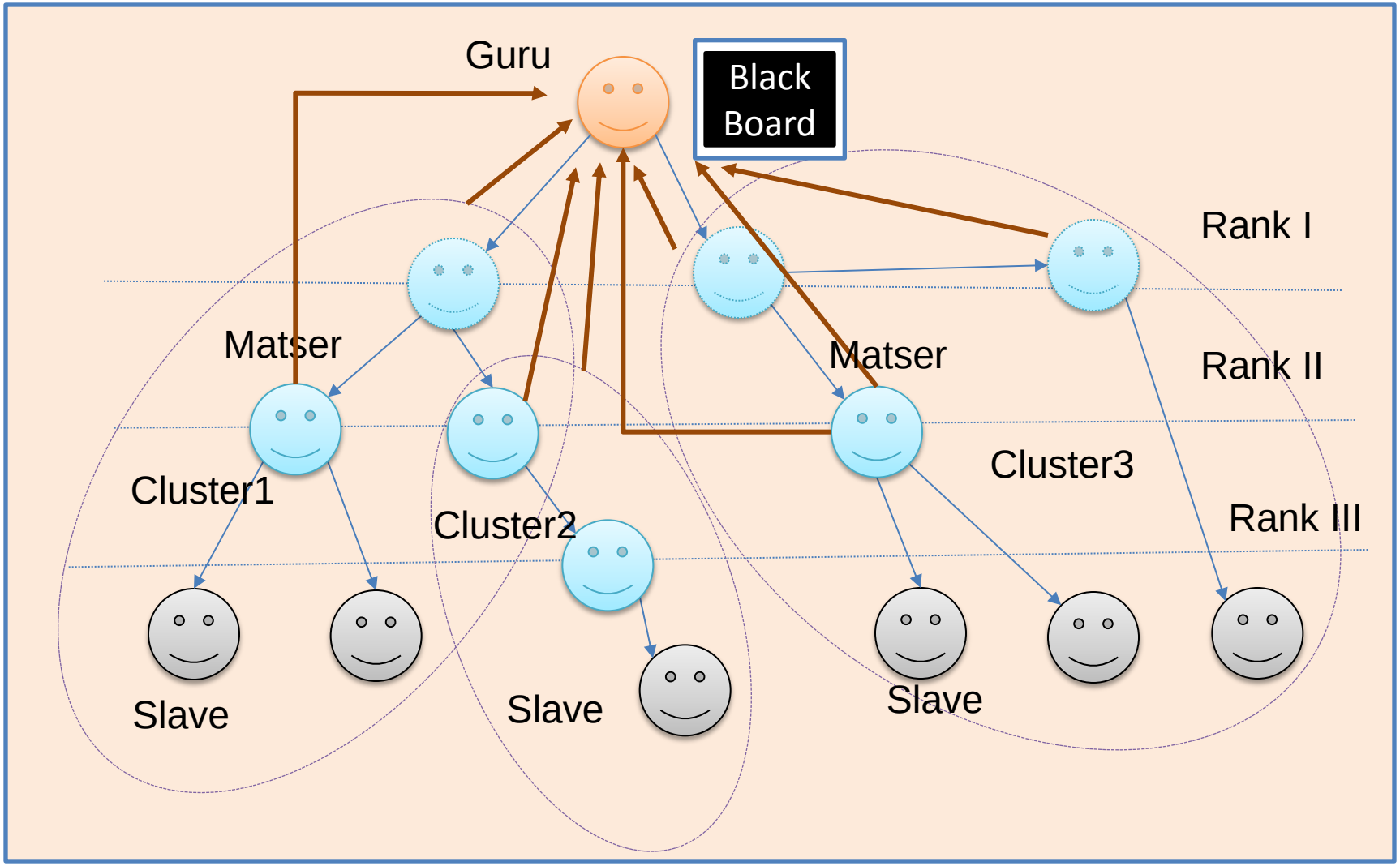
The advanced architecture of the agents

- Agent is multi-threaded and multi-tasking.
- Communication manager (CM) manages the input and output of queries and answers:
  - When it receives a query q and $q \in Oi$, a new goal is added to the goal list: $Gi := Gi \cup \{ <q,Aj, tf >\}$.
  - When it receives an answer, it sends it to the integration machine.
  - It sends the answers and queries to the other agents, following the correct protocols and reporting all the activity.
- The reasoning engine receives as inputs fact values and performs a specialization cycle: $S : KB \times f \rightarrow KB'$ is a data-driven process that begins when the input is a new fact value f. This triggers a complete specialization process over the KB and a new specialized KB' is generated.

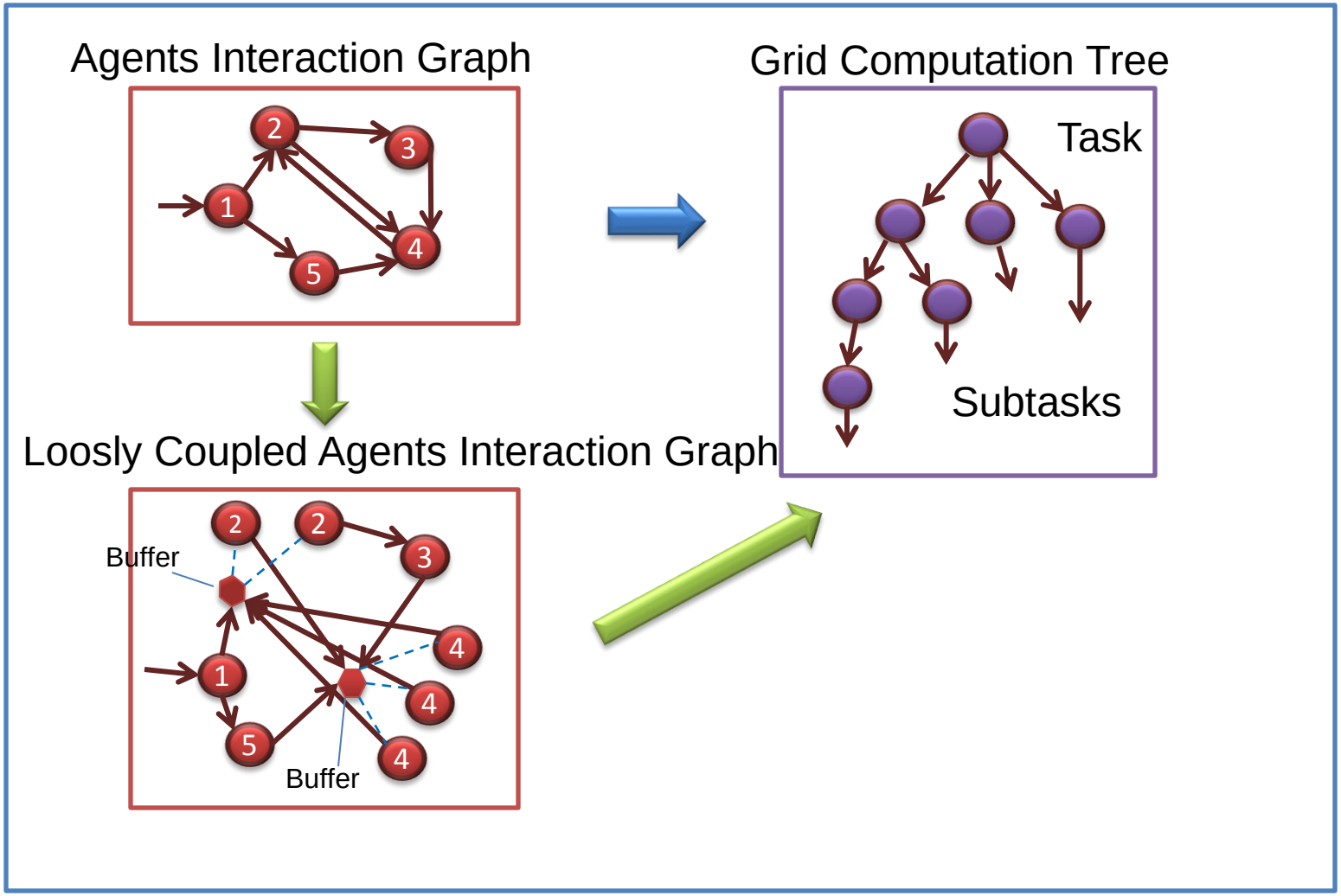
The advanced architecture of the agents

- The integration machine receives as input a complete answer (facts and eventually a set of rules) and incorporates them into the KB.
- The broker machine receives as input a trigger signal indicating:
  - A goal deadline ends. If the goal doesn't have a definitive value, then the answering machine has to elaborate other kinds of answers
  - The definitive value for a goal is found, and then the obvious response is the definitive value.
- The evaluator machine is a goal-driven process $I : KB \times g \rightarrow g^*$ that begins when the agent process a goal g. It triggers a complete exploring process obtaining a set of new goals g*, which are necessary to find values of g with better quality
- The blackboard is a decision making machine. The Agents the products (answers) are placed in a buffer – blackboard and owner agent coordinates a decision making process

New hierarchical classification schema of agents



Projection MAS Into Grid

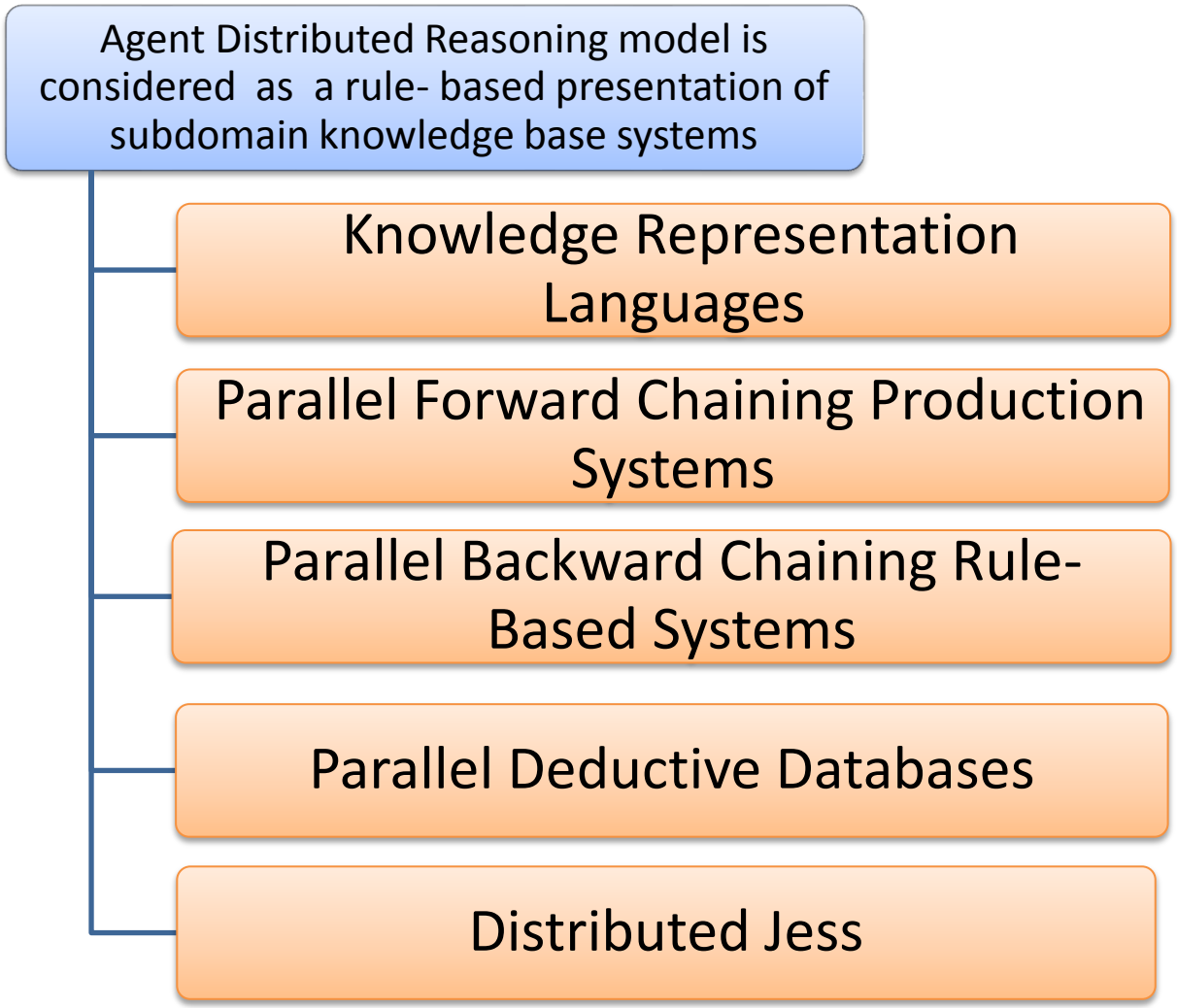


Stigmergic agent system (SAS)

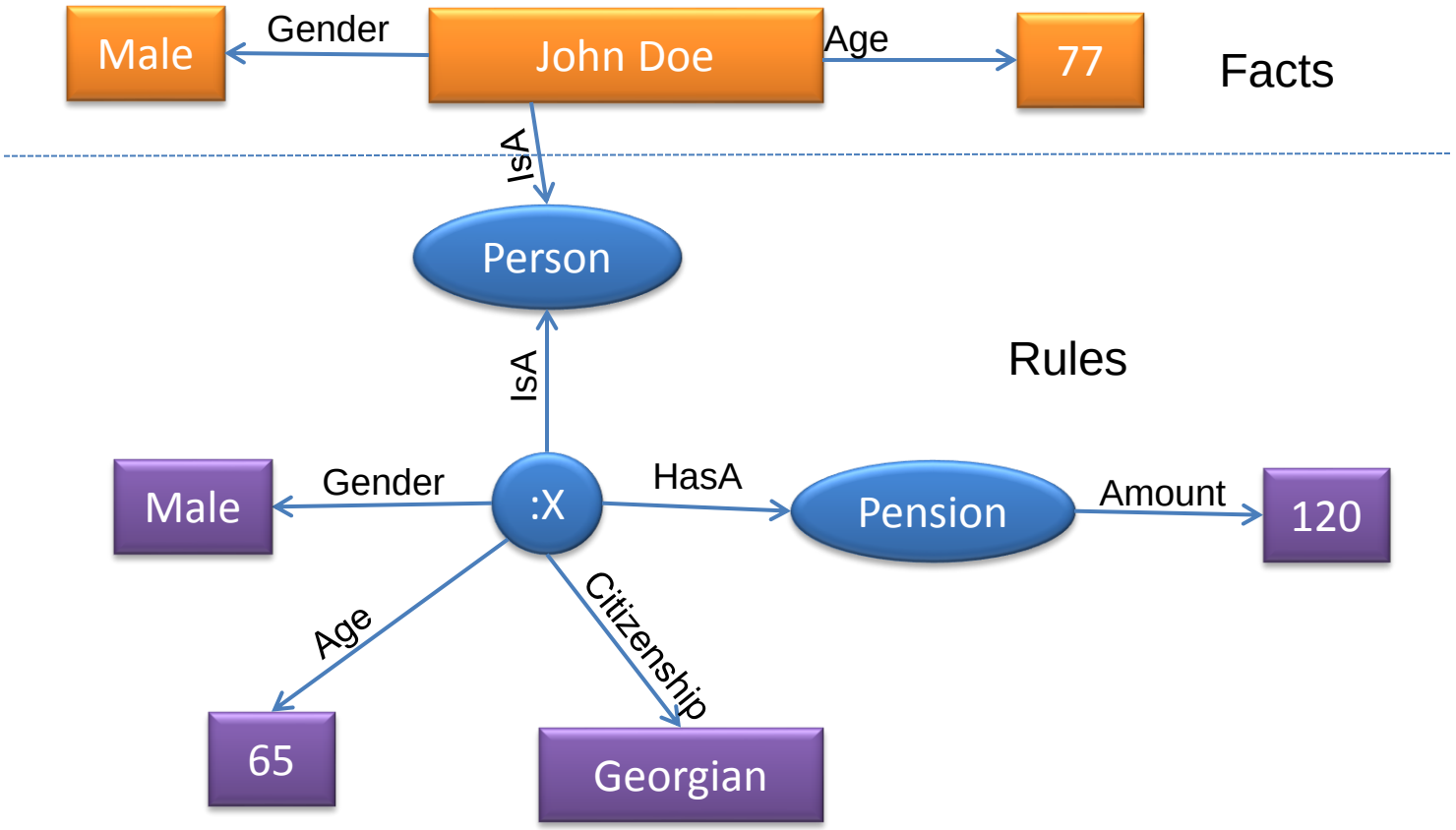
Stigmergy provides a general mechanism that relates individual and colony level behaviors: individual behavior modifies the environment, which in turn modifies the behavior of other individuals . The SAS mechanism solve next problems

- direct communication: agents are able to exchange different types of messages in order to share knowledge and support direct interoperation; the knowledge exchanged refers to both local and global information
- indirect (stigmergic) communication: agents have the ability to produce pheromone trails that influence future decisions of other agents within the system

Distributed Reasoning

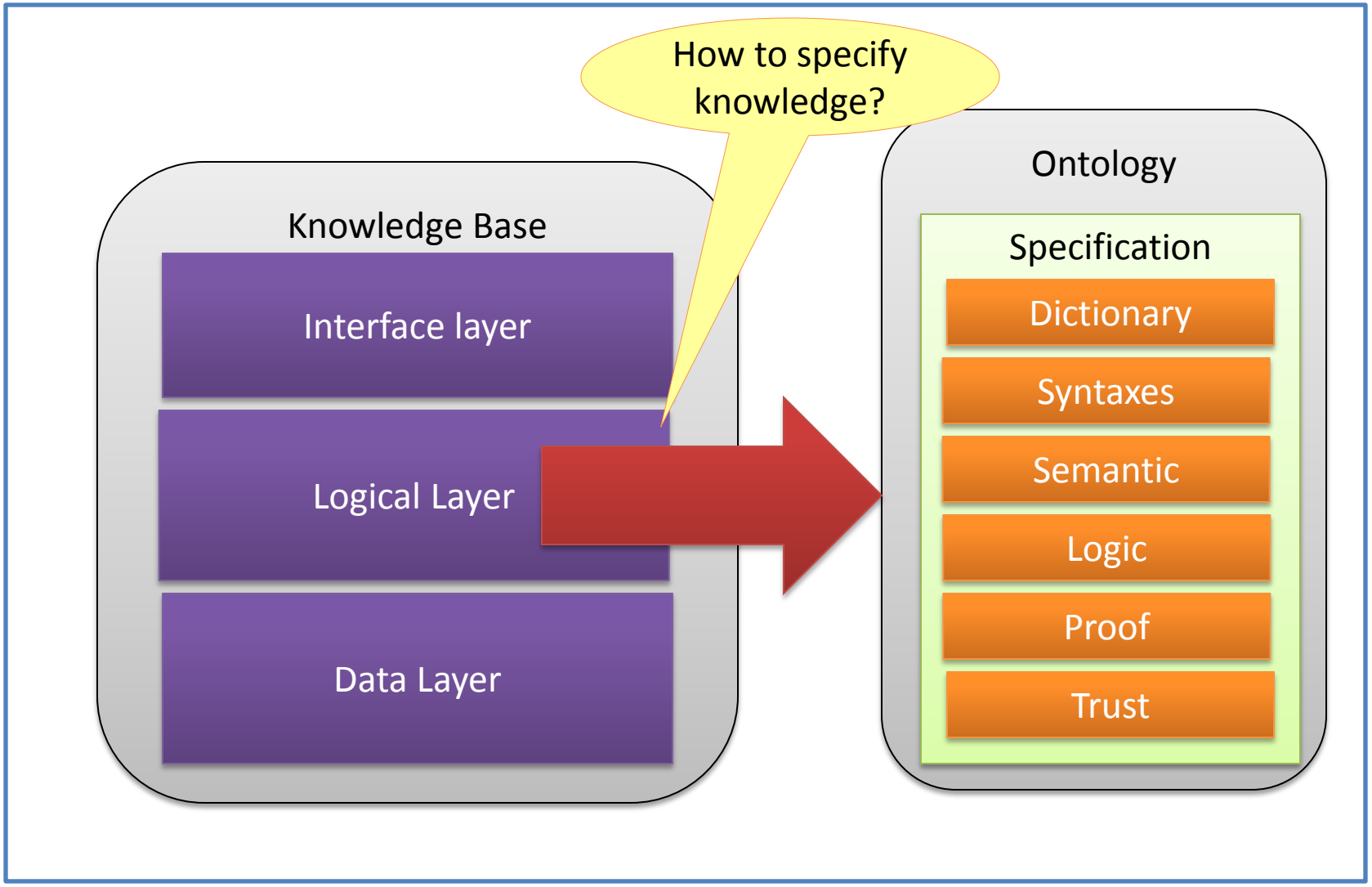


Knowledge Representation Languages



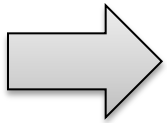
En example presentation of knowledge as semantic network

Knowledge Representation and Ontology



Design Decisions: Categorical Presentation

Categorical logic sentence
"subject–copula–predicate"
"S → P" ("S ↔P")

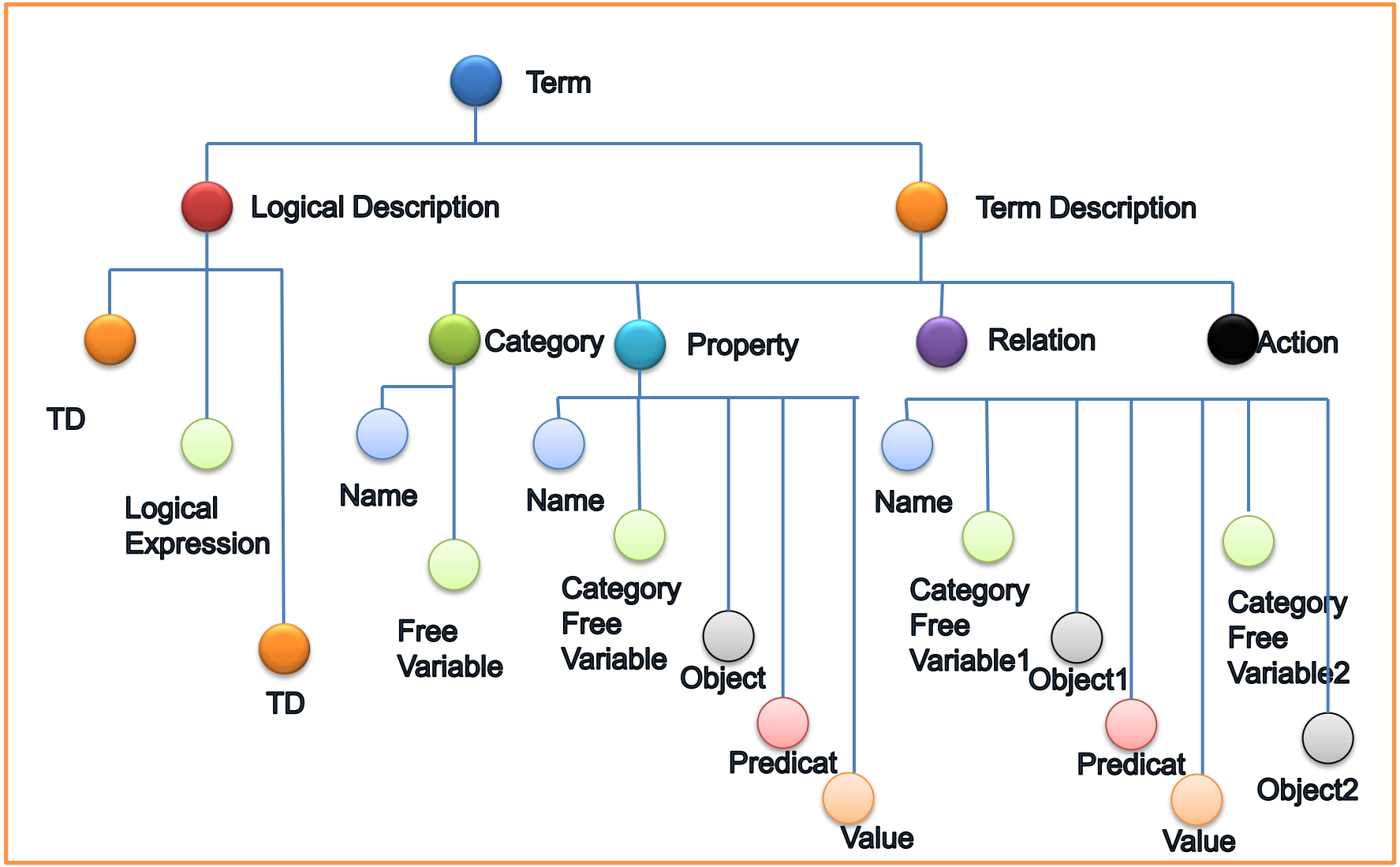


"if - then"

$$A_1,A_2,..,A_n \xrightarrow{R} D(C[\alpha_2, .., \alpha_m]), \quad \{v, b\} \qquad (5)$$

$A_1,A_2,..,A_n$ - elementary or compound terms or even logical expressions
R - is a relation
D -is a domain ,
C - is a category of the conclusion which belongs to D
$\alpha_2, .., \alpha_m$ – are terms which express the values of the attributes and relation of the category *C*
$v,b \in [0,1] \subset R$ - indicates the degree of the truth-value (frequency and confidence).

The Term Syntaxes



Categorical Rule Based Presentation (Examples)

Validate Benefit : State Pension: State Age Pension

N	Attribute	Pre-condition	Possible Value	Validation Condition
1	Applicant Gender		Man/ Woman	Man/ Women
2	Citizenship		Georgia, Other	Georgia
3	Age		Number	>60 when applicant is a woman; >65 when applicant is a man;
4	Civil Servant		Yes/no	Yes/no
5	Applicant works as a teacher or science	Actual if “4” response is “yes”	Yes/no	Yes
6	Civil Register Status		Active/dead	Active

Categorical Rule Based Presentation (Examples)

```
then Benefit [:x]
then State_Pension [:y, isA (:y, Benefit)]
then Age_Pension [:z, isA (:z, State_Pension)]
If  (Applicant_Gender (:x1,"man") XOR Applicant_Gender (:x1,"woman")) AND
    Citizenship((:x1,"Georgia") ) AND
    (Applicant_Gender (:x1,"man") -> Age(:x1,65,">")) AND
    (Applicant_Gender (:x1,"woman") -> Age(:x1,60,">")) AND
    (Civil_Servant (:x1,"yes") XOR Civil_Servant (:x1,"no")) AND
    (Civil_Servant (:x1,"yes") ->
       Applicant_Works_As_A_Teacher_Or_Science (:x1,"Yes")) AND
    Civil_Register_Status(:x1,"Active") )
then
       State_Age_Pension [ :y1,
                           Person[:x1]
                         ]
```

Rule Based Presentation and Reasoning
Categorical Presentation. Description of semantics

S4: if $A \xrightarrow{R} B$ is *true* and $B \xrightarrow{R} C$ is *true* and relation R is a transitive an reflexive , then $A \xrightarrow{R} C$ is *true,* that maps the transitivity condition

If A ,B,C are numbers and $A < B$, $B < C$ *then* $A < C$

S5: if $x \xrightarrow{IsA} D$ and $\rightarrow p(D,\_)$ are *true* , then $\rightarrow p(x,\_)$, this is famous "modus ponens" rule

If Socrat is a human , human is a mortal then Socrat is a mortal

Rule Based Presentation and Reasoning
Categorical Presentation. Description of semantics

It can be easily shown that representation allows predicates of higher tier to be represented. E.g. the phrase: 'I know that the Earth is round and turns around' may be written in the following manner.

P("I") $\xrightarrow{know}$
Earth(:x){round(:x),turns_around(:x)}

Theorem 1: The expression

$$A_1 \otimes A_2, A_1 \Rightarrow A_3, A_2 \Rightarrow A_4 \xrightarrow{R} C$$

is a equivalent to the expression .

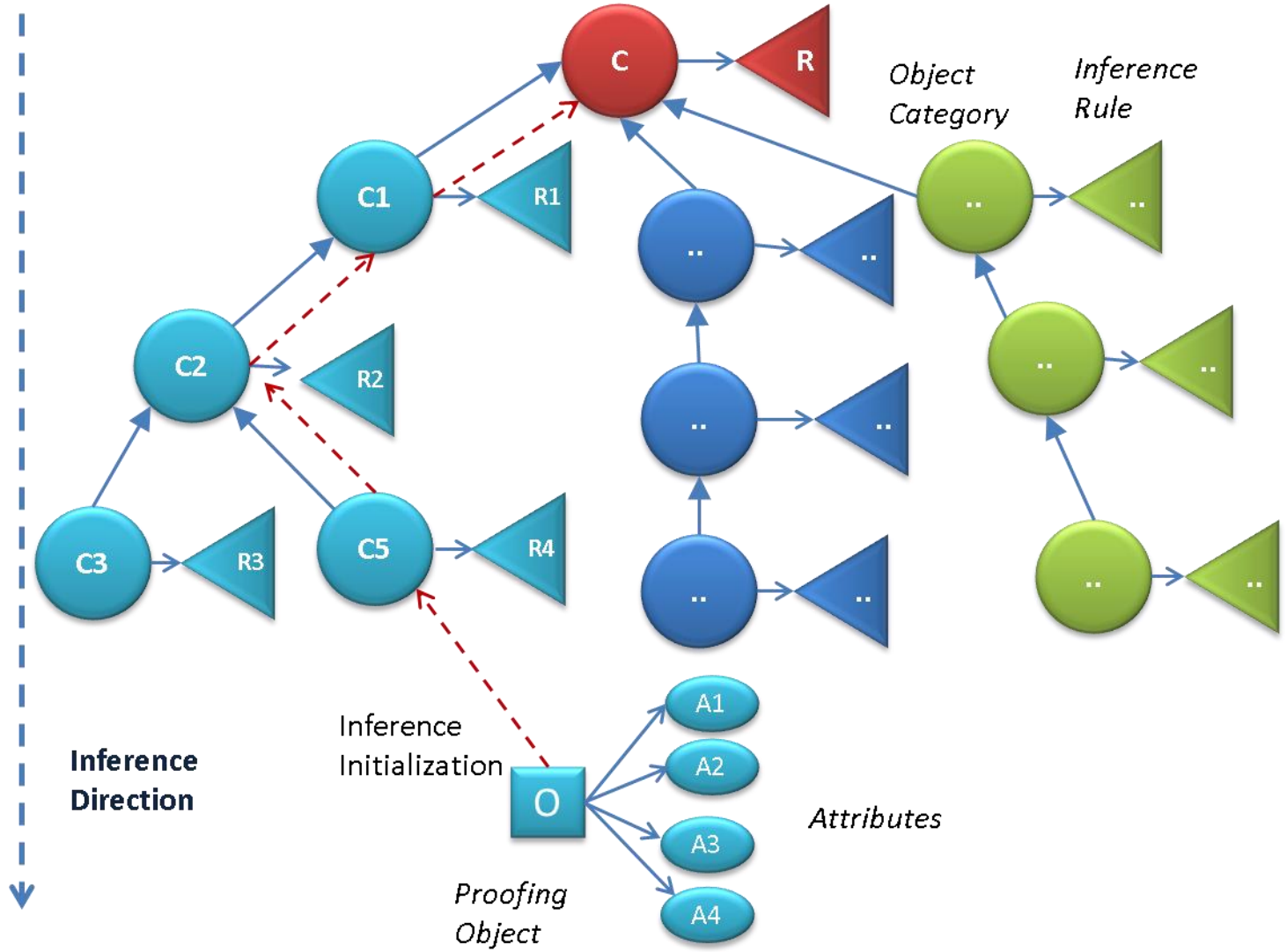
$$A_1, A_3 \xrightarrow{R} C,\ A_2, A_4 \xrightarrow{R} C$$

Rule Based Presentation and Reasoning
Categorical Presentation. Reasoning

The possible form of queries

	Request form	Reasoning direction
1	$A_1,..,A_n \rightarrow ?$	Direct
2	$A_1,..,A_n \rightarrow C\{?,p(?),r(?,\_)\}$	Direct Back Chaining
3	$? \rightarrow B$	Back Chaining
4	$?,A_1,p(?),r(?,\_) \rightarrow B$	Back Chaining
5	$(A_1,..,A_n \rightarrow B)?$	Direct Back Chaining

Rule Based Presentation and Reasoning
Categorical Presentation. Reasoning

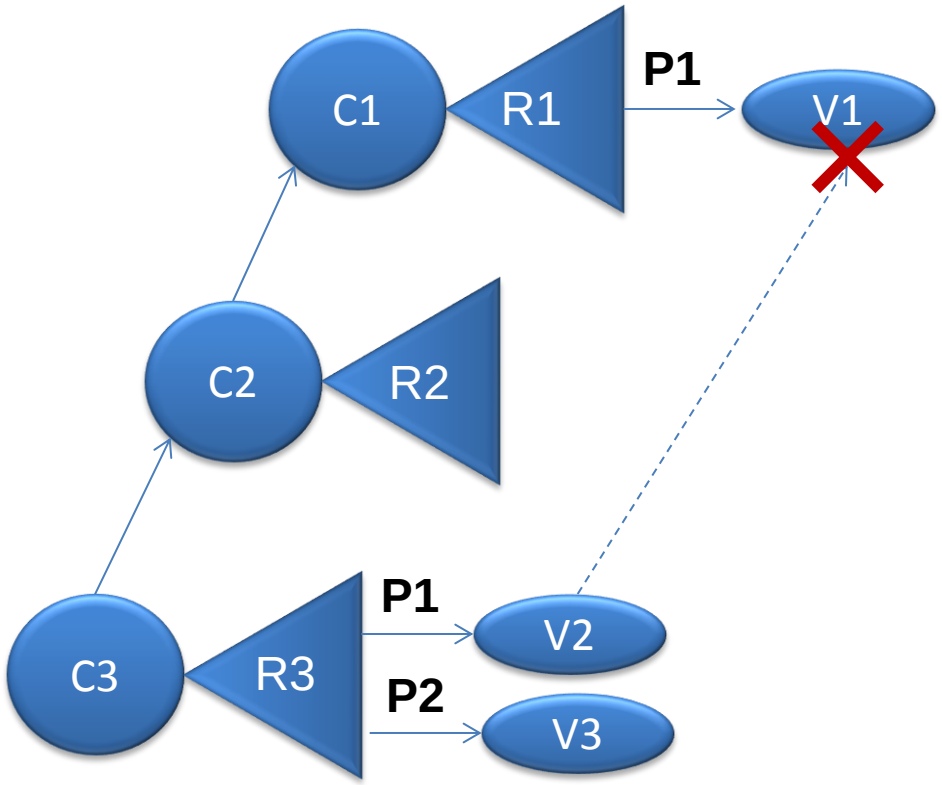


Rule Based Presentation and Reasoning: Exclusion

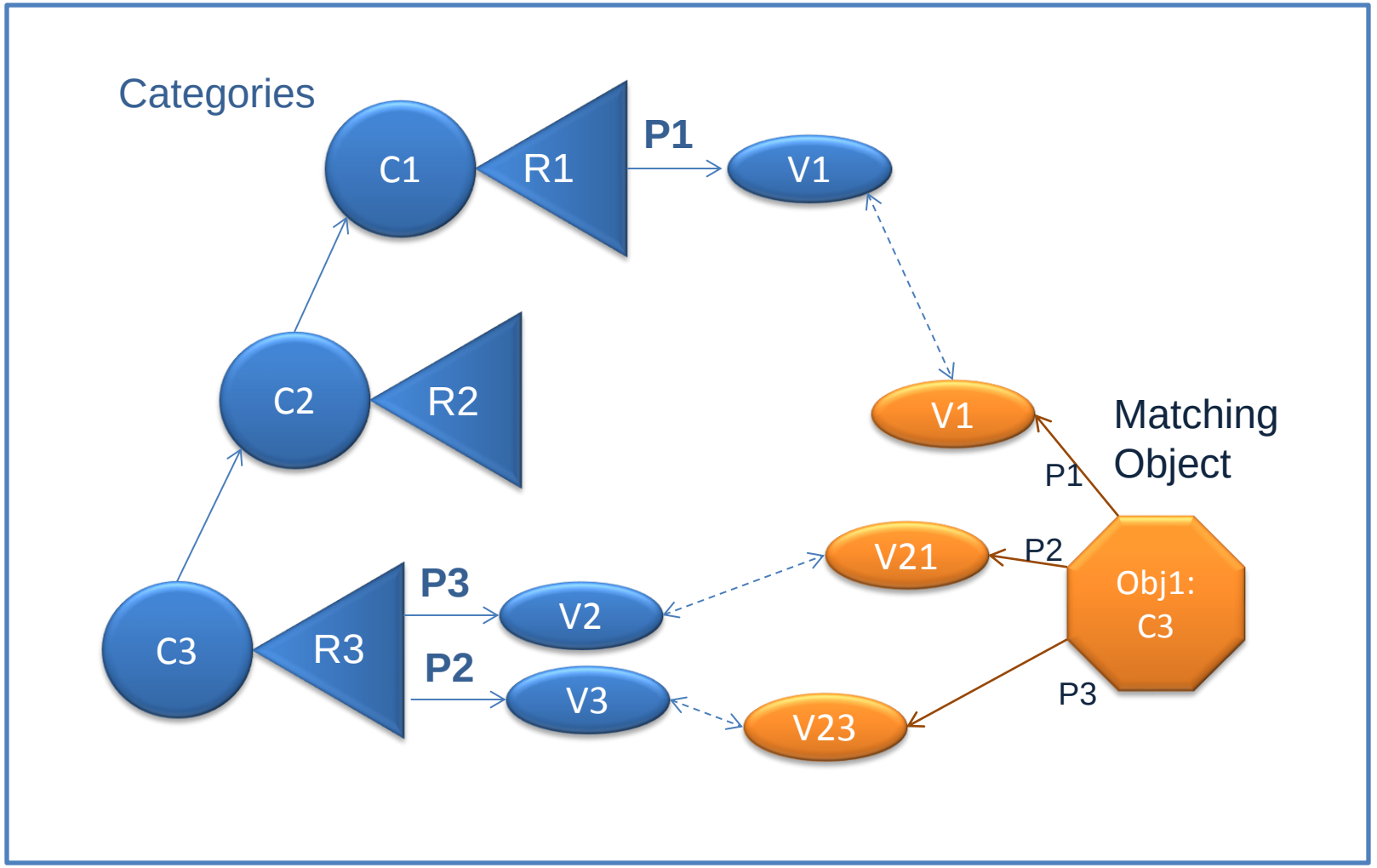
"Bird can fly."

"Penguin is a bird."

"Penguin can not fly."



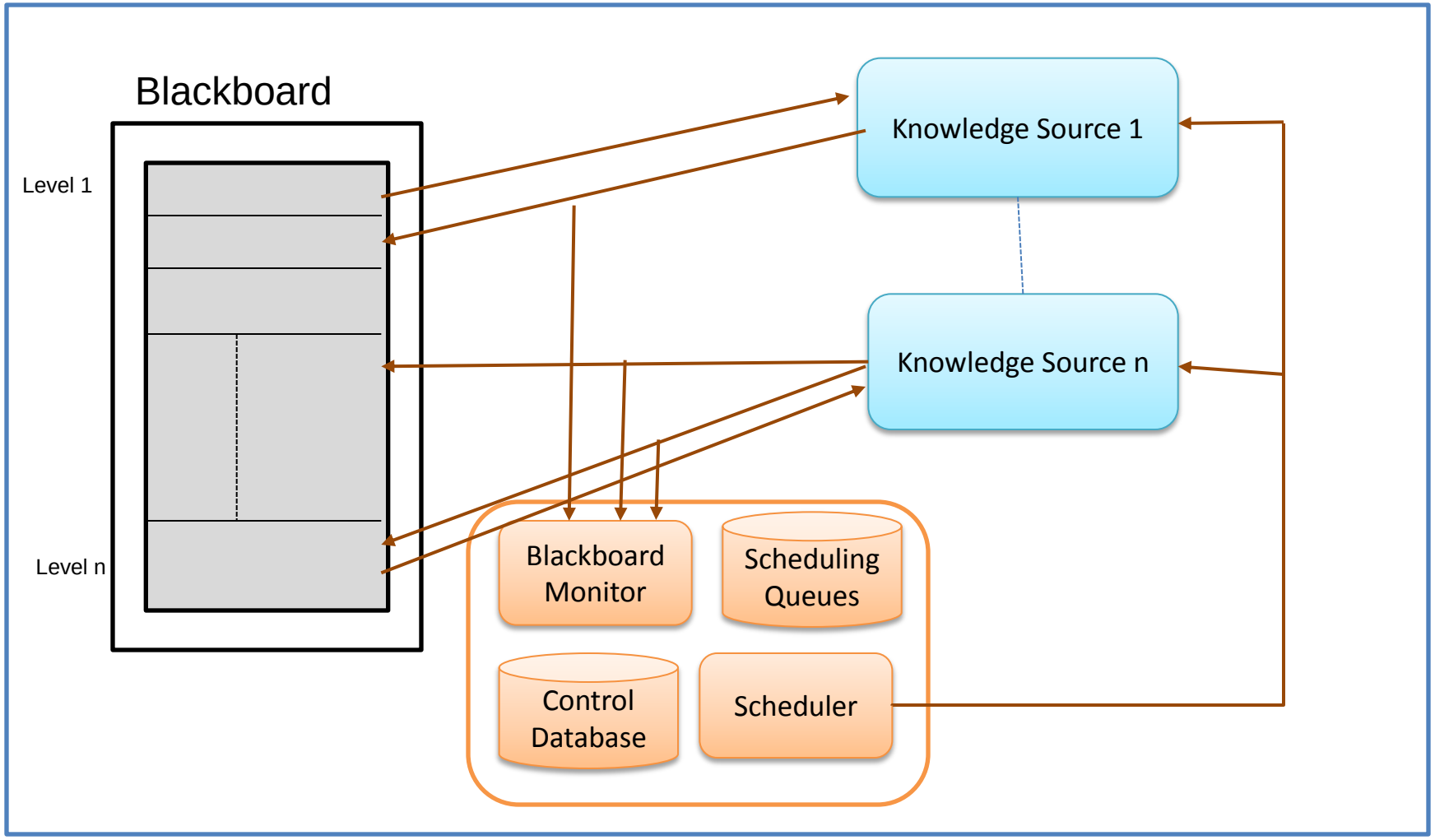
The Distributed matching Algorithm of Reasoning: Matching Schema



The Distributed matching Algorithm of Reasoning: Matching Algorithm

```
     var object input
1.   var AttributList := getObjectAttributeList(object)
2.   var ObjectCategory := getObjectCategory(object)
3.   var CategoryList := getParentCategories(object, Ord_top_down)
4.   var flgSuccess  := false
5.   loop  var category  in CategoryList
6.       loop var  attribute in AttributList
7.           if  category.Match(attribute) then
8.               AttributList.remove(attribute)
9.               if  AttributList.isEmpty() then
10.                  flgSuccess := true
11.                  break;
12.              end
13.          end
14.      end loop
15.      if flgSuccess == true then break;
16.  end loop
```

The blackboard architecture



Argumentation Frameworks

Definition 1 An *argumentation framework* is a pair

AF = <AR,attacks>

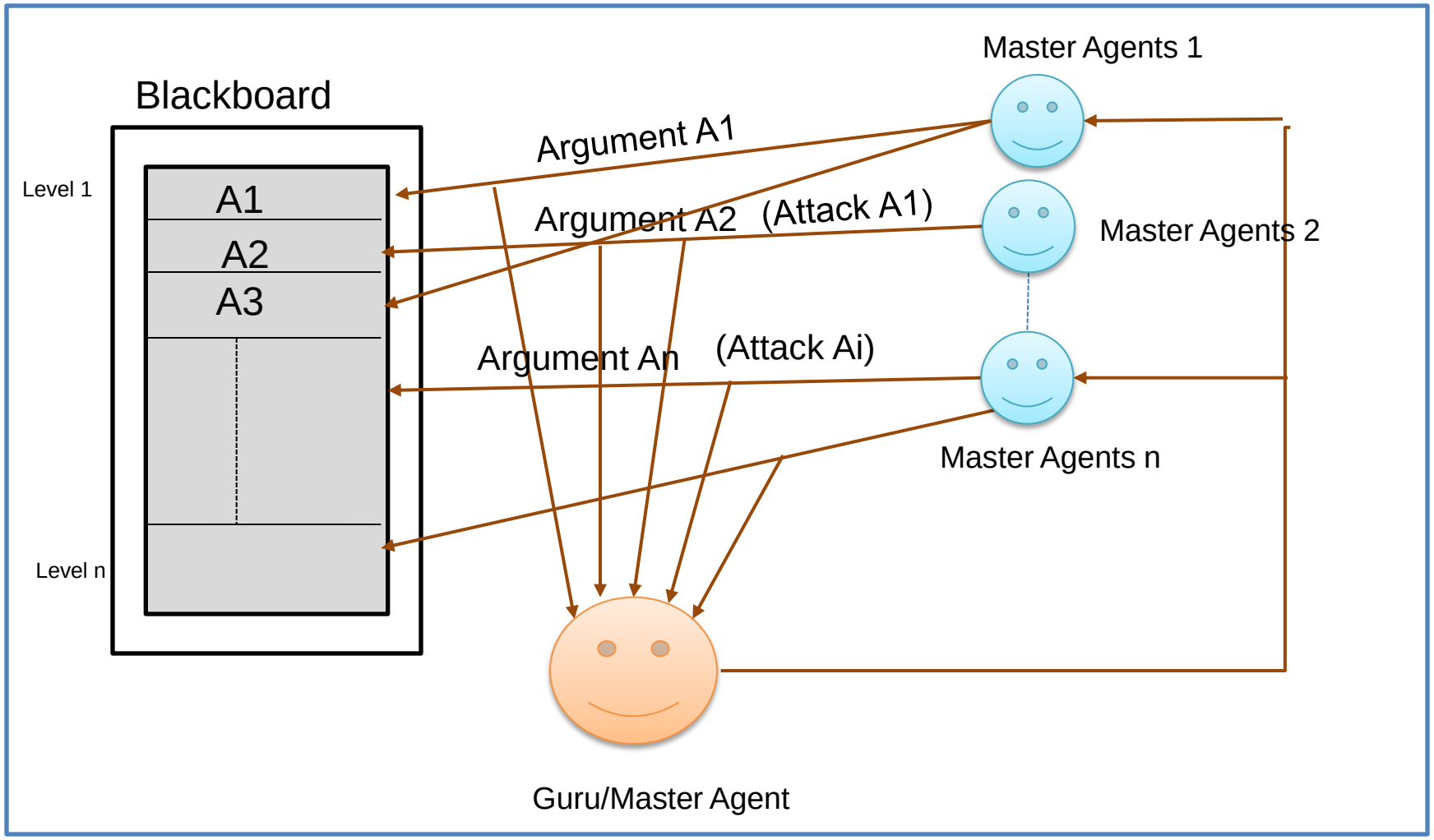
where AR is a set of arguments, and attacks is a binary relation on AR, i.e. attacks $\subseteq$ AR × AR.

Definition 2 A set S of arguments is said to be *conflict-free* if there are no arguments A,B in S such that A attacks B.

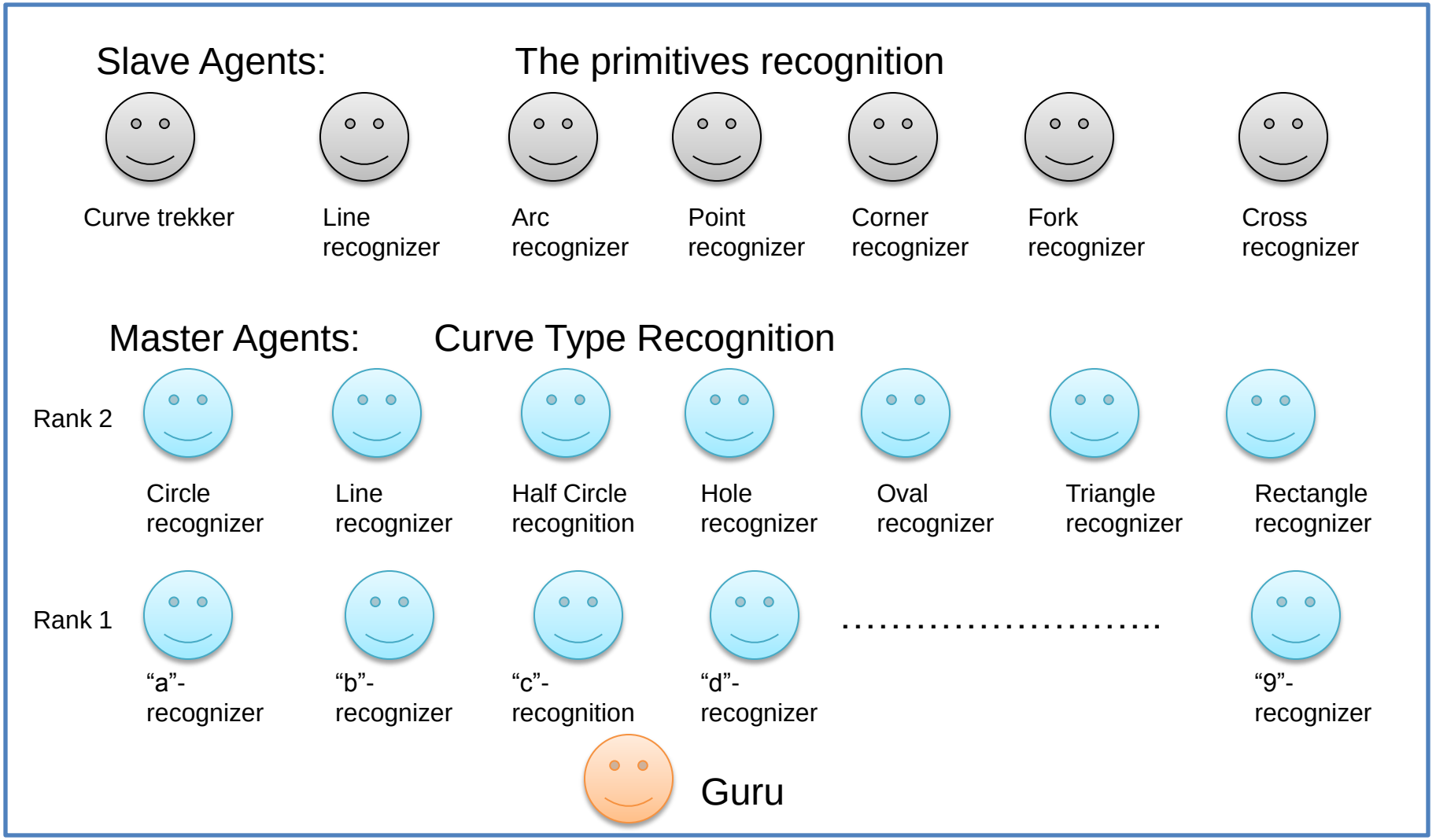
Definition 3 (1) An argument A $\in$ AR is *acceptable* with respect to a set S of arguments iff for each argument B $\in$ AR: if B attacks A then B is attacked by S.
(2) A conflict-free set of arguments S is *admissible* iff each argument in S is acceptable wrt S.
(3) S is a *preferred extension* if it is a maximal (with respect to set inclusion) admissible subset of AR.

The key notion here is the *preferred extension* which represents a consistent position within AF, which is defensible against all attacks and which cannot be further extended without becoming inconsistent or open to attack.

Agent-based Argumentation + Blackboard architecture



Example: Manuscript recognition system



Example: Manuscript recognition system

Knowledge Base for the master agent

1. Line(x),Line(y), Equal(End(x),End(y)), Parallel(x,y) -->
Line(z){Start(z)=Start(x), End(z)=End(y) } - rank 2

1. Line(x),Line(y), Cross(x,y) --> Symbol (z) {Name(z,"x")} - rank 1
2. Line(x),Line(y), Cross(x,y), Not Vertical(x) --> Symbol (z) {Name(z,"x")} - rank 2

1. Line(x),Line(y), Cross(x,y) --> Symbol (z) {Name(z,"+")} - rank 1
2. Line(x),Line(y), Cross(x,y), Vertical(x) --> Symbol (z) {Name(z,"+")} - rank 2
3. Line(x),Line(y), Cross(x,y), Horizontal(y) --> Symbol (z) {Name(z,"+")} - rank 2

Related Works

1. T. Bench-Capon, 'Value based argumentation frameworks', in *Proceedings of Non Monotonic Reasoning*, pp. 444–453, (2002).
2. P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995
3. Costin B dic 1, Lars Braubach2, and Adrian Paschke, Rule-based Distributed and Agent Systems
4. Jiefei Ma, Distributed Abductive Reasoning System and Abduction In the Small, Computing Final Year Project Report, June 22, 2007, Imperial College of Science, Technology and Medicine University Of London
5. Elth Ogston , Benno Overeinder, Maarten van Steen, and Frances Brazier, A Method for Decentralized Clustering in Large Multi-Agent Systems
6. Zhang, T., Ramakrishnan, R., Livny, M.: BIRCH:A New Data Clustering Algorithm and Its Applications. Data Mining and Knowledge Discovery, 1(2) (1997) 141–182
7. Fadi ABDIN, Angel OSORIO , AN EFFICIENT ANT COLONY OPTIMIZATION CLUSTERING ALGORITHM , Ubiquitous Computing and Communication Journal , Volume 6 Number 4

Future Works

- Choice Learning Model
- Development of Learning Module
- Enhance Engine Reasoning Performance

Thank you for your attention

Any Questions?

Contact details: zbossik@gmail.com